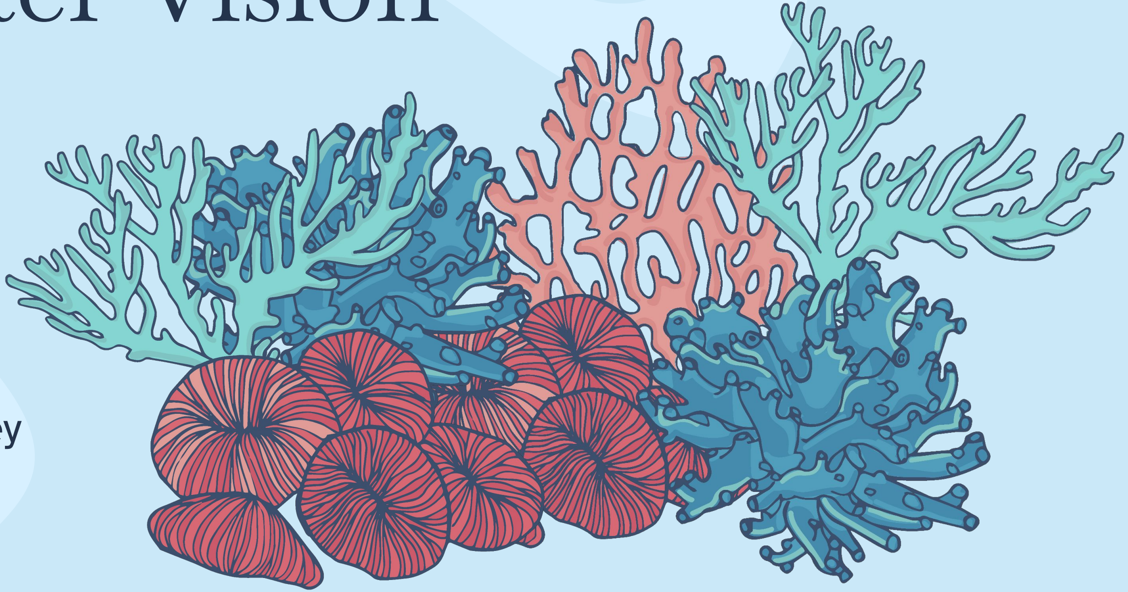
Reef Madness: Coral Health Monitoring via Computer Vision



Jaskirat Kaur, Qilin Zhou, Grey Xu

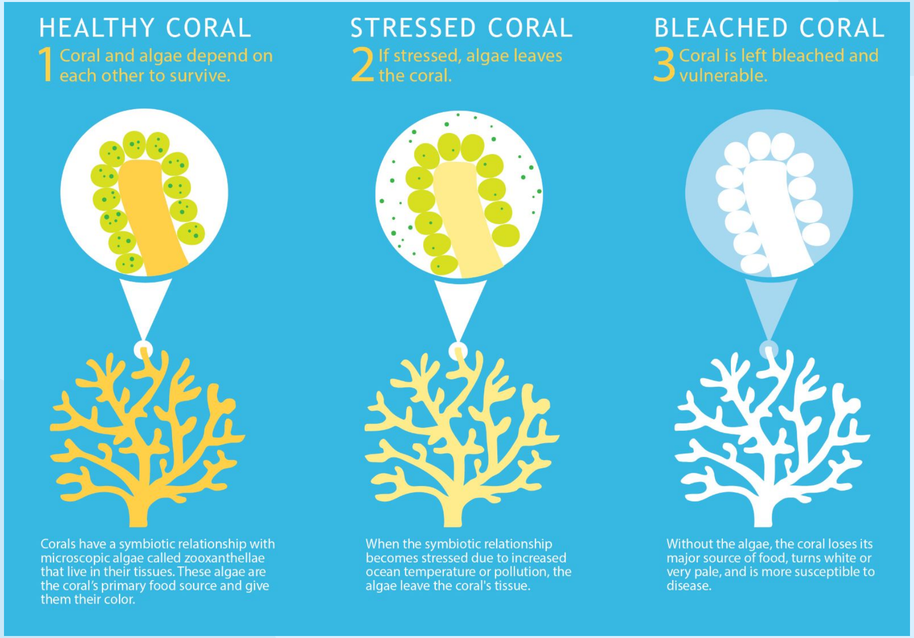
Table of contents

01

Background

Mass Bleaching — A Coral Reef Crisis

- **Coral Bleaching**: Triggered by warm water, corals expel algae, leading to a white appearance and increased mortality risk.
- **Climate Change**: In 2005, half of the U.S. Caribbean coral reefs were lost due to severe thermal stress, surpassing previous two decades.
- **Goal**: Using computer vision, we aim to detect coral bleaching and create generative images to illustrate potential impacts.



Research Questions

Classification

- **Healthy Status Classification**: Determine the health of corals .
- ***Species Identification:** classify different coral species

Generation

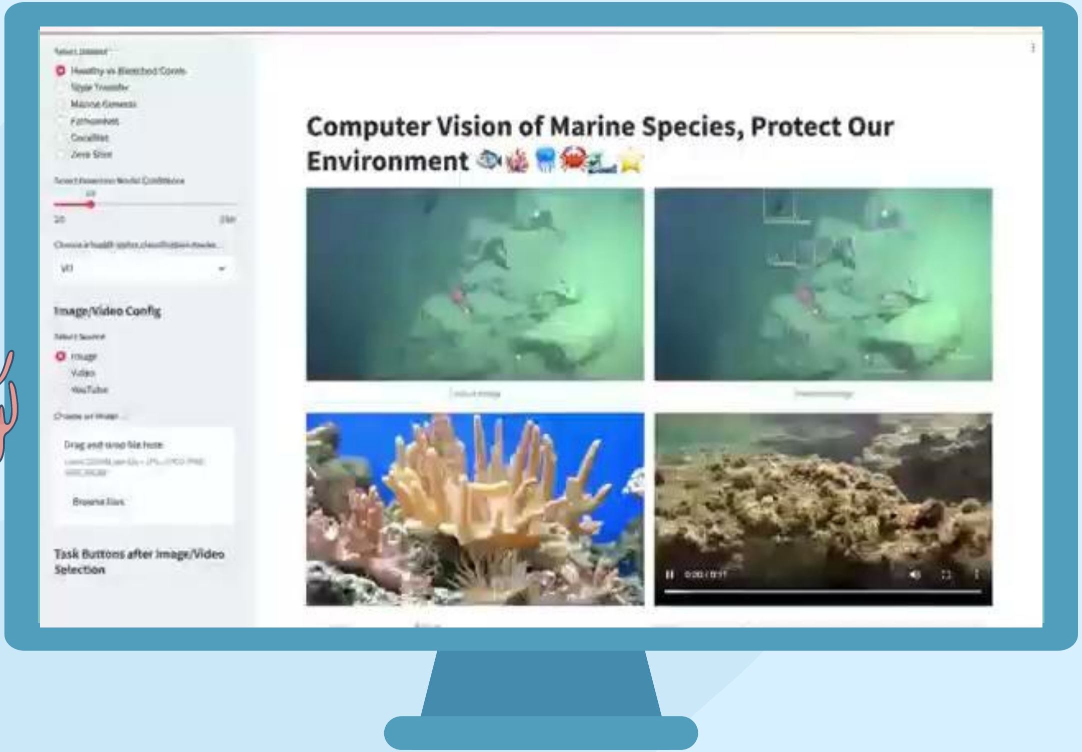
- **Image Generation**: Visualize potential unhealthy states of currently healthy corals.
- ***Style transformation**: Create stylized, cartoon-like images of corals.

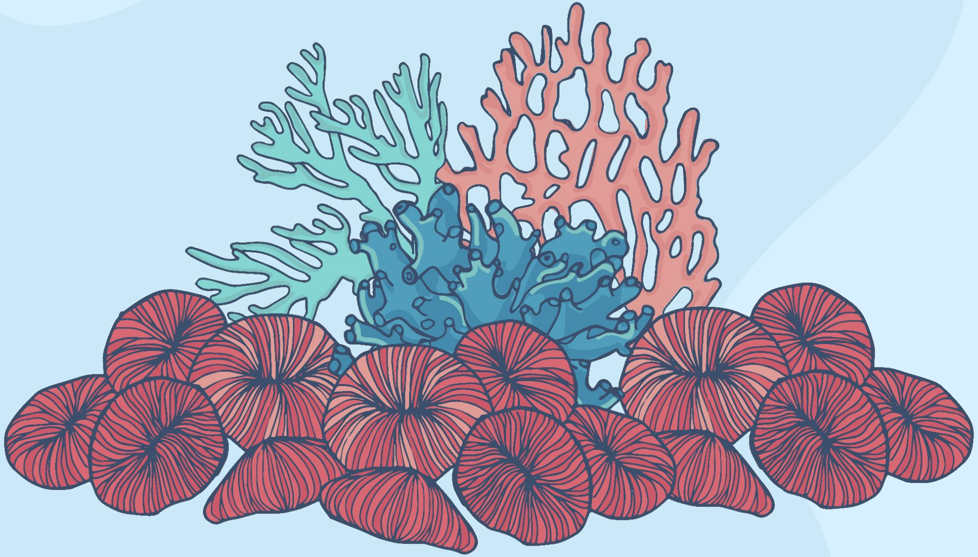
*Detection

- ***Species Detection**: Identify corals and other marine species in the images.

* means additional question that is not fully discussed in this presentation

Product Demo





02

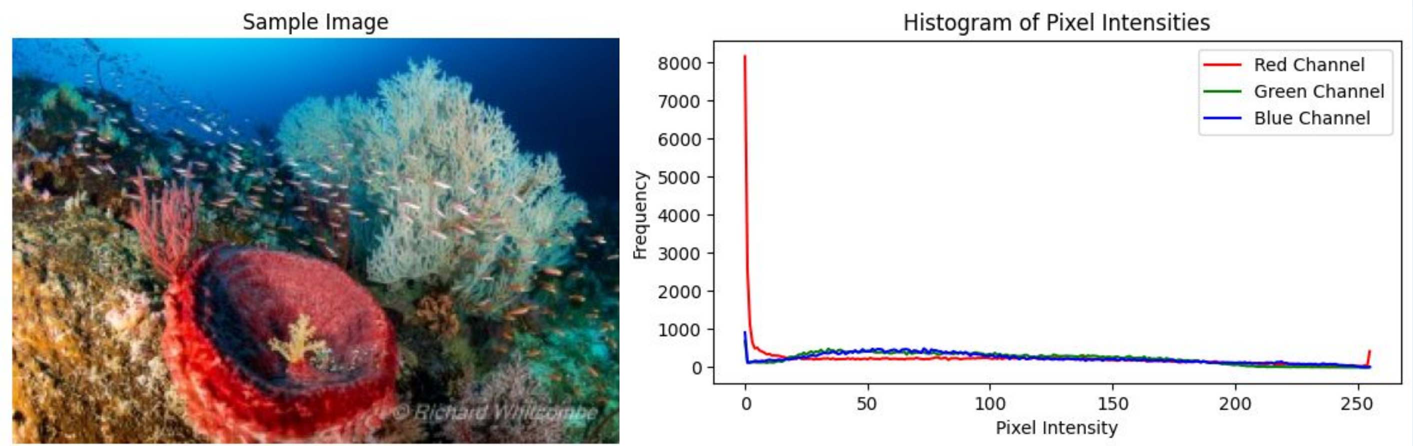
Datasets

Dataset Overview

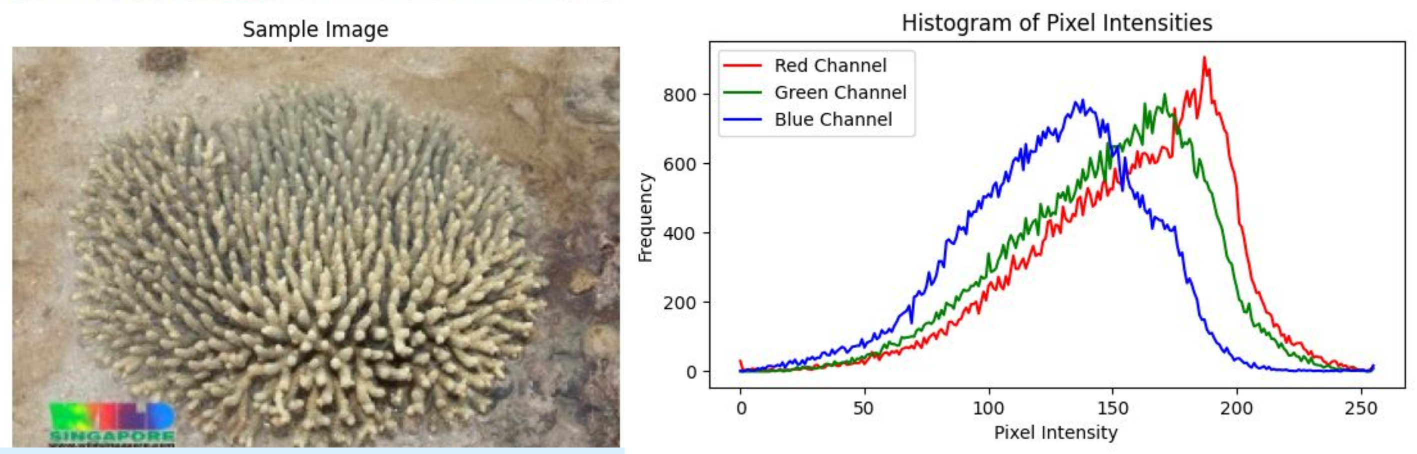
	Coral Health	Fathom Net	Coral Net	Marine General
Question	Classification/ Generation	Detection	Classification	Detection
Objects	Coral Health Status	Coral Reefs Bounding box	Coral Species type	Marine ecosystem Bounding box
Sizes	900	24,000+	300,000+	798
Source	Flickr API	Fathom Net	Coral Net	Kaggle

Coral Health

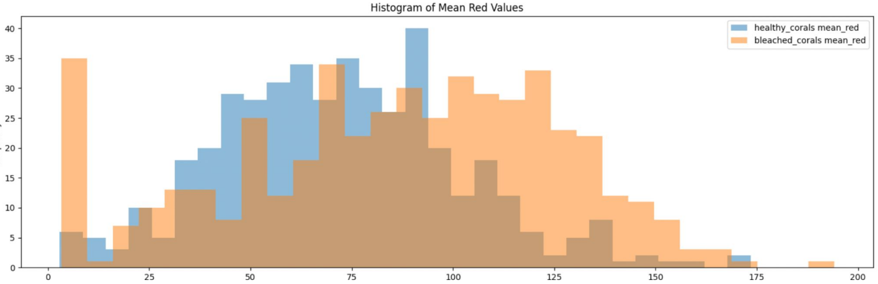
Healthy Coral

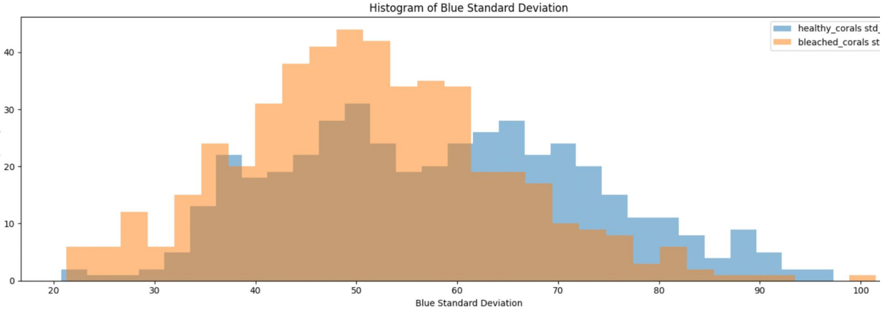


Bleached Coral



Exploratory Analysis - Coral Health





Logit Regression Results

Dep. Variable:	label_encoded	No. Observations:	923
Model:	Logit	Df Residuals:	914
Method:	MLE	Df Model:	8
Date:	Sat, 27 Apr 2024	Pseudo R-squ.:	0.2050
Time:	21:41:02	Log-Likelihood:	-507.66
converged:	True	LL-Null:	-638.58
Covariance Type:	nonrobust	LLR p-value:	5.332e-52

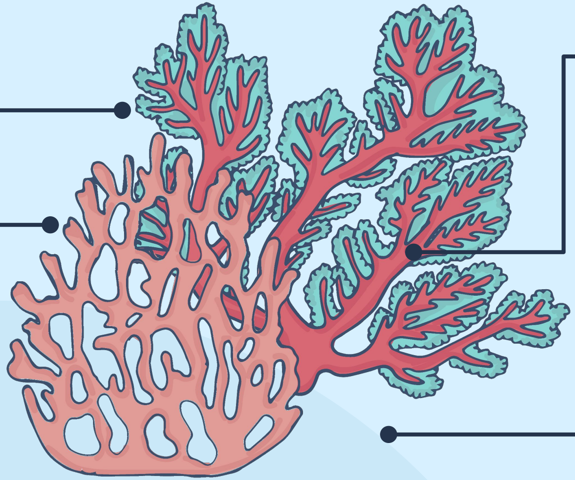
	coef	std err	z	P>|z|	[0.025	0.975]
const	-0.1204	0.621	-0.194	0.846	-1.337	1.097
mean_red	0.0167	0.003	5.495	0.000	0.011	0.023
mean_green	-0.0050	0.006	-0.871	0.384	-0.016	0.006
mean_blue	0.0033	0.005	0.691	0.490	-0.006	0.013
std_red	-0.0536	0.007	-7.796	0.000	-0.067	-0.040
std_green	0.1175	0.013	8.780	0.000	0.091	0.144
std_blue	-0.0758	0.010	-7.379	0.000	-0.096	-0.056
width	0.0061	0.001	4.242	0.000	0.003	0.009
height	-0.0084	0.002	-4.508	0.000	-0.012	-0.005

Fathom Net API

Open-source image database for understanding our ocean and its inhabitants

109871 Images

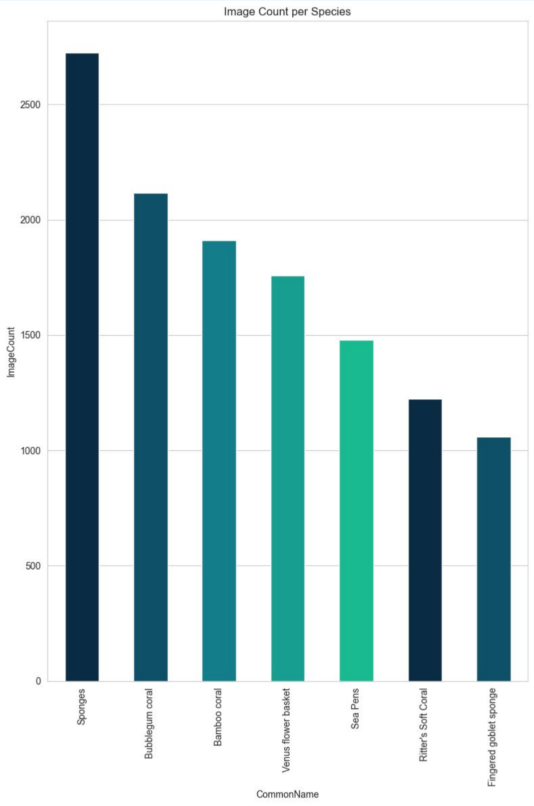
MetaData

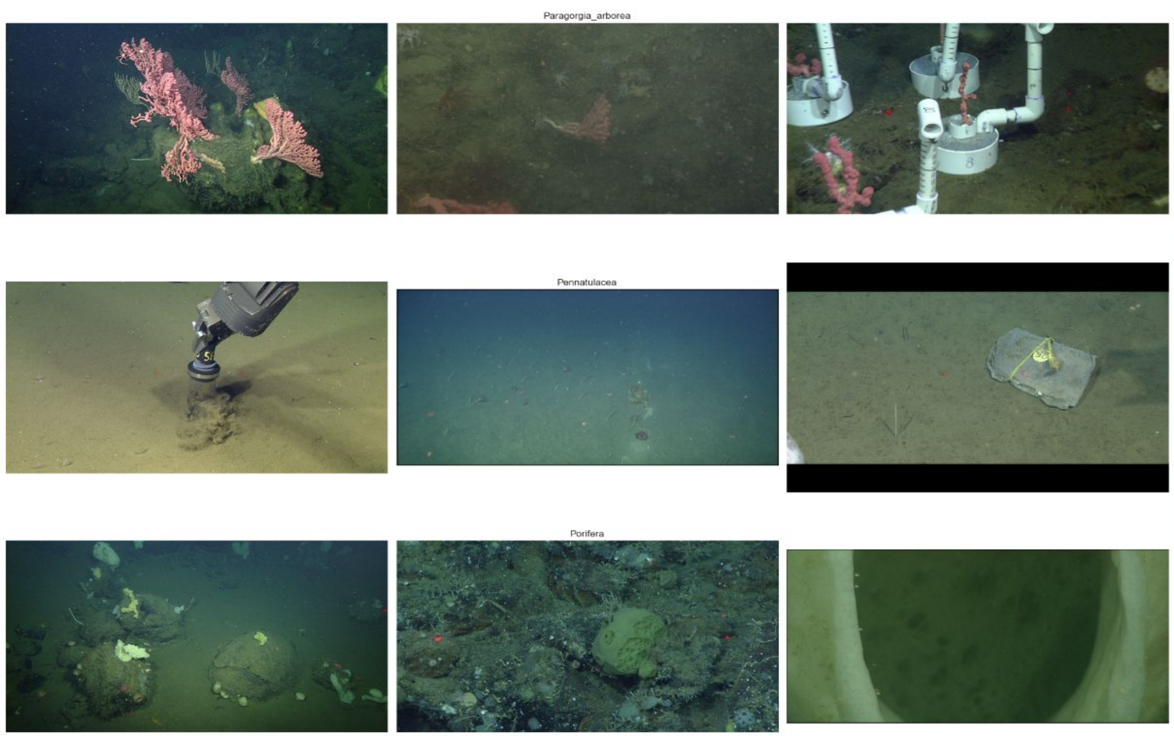


Bounding Boxes

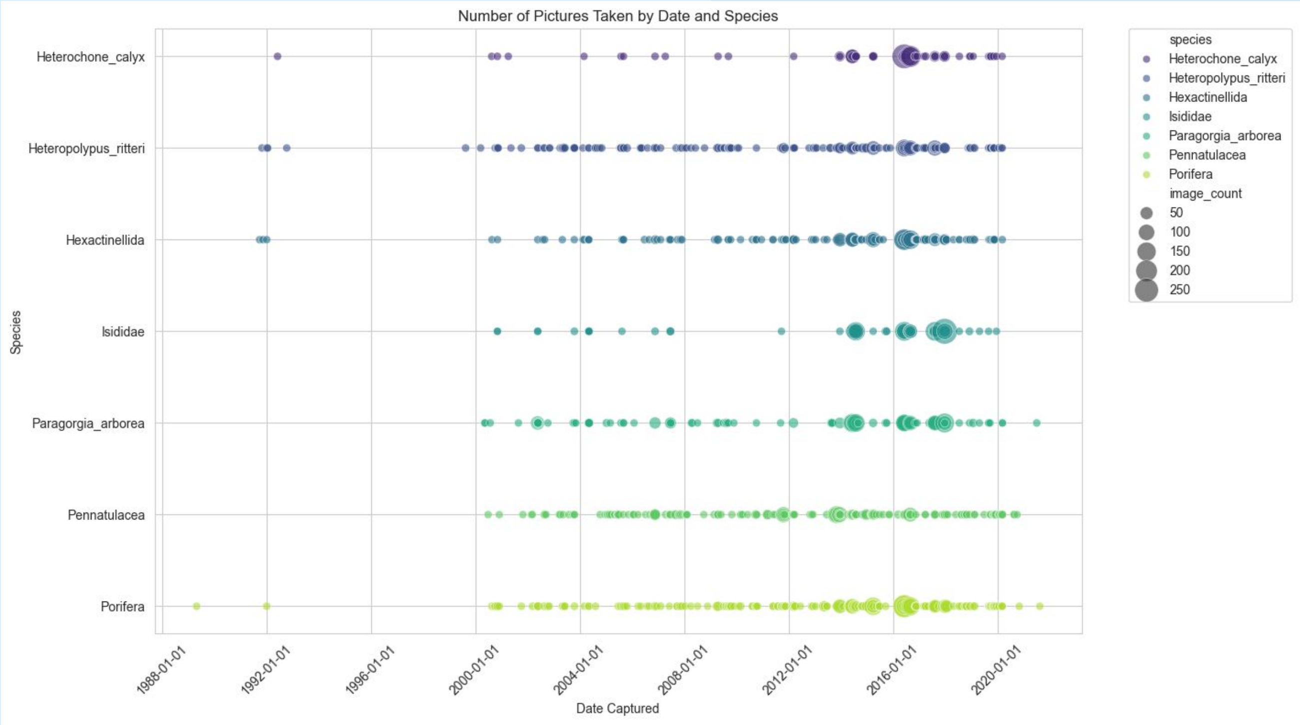
Geospatial Data

Exploratory Analysis - Fathom Net





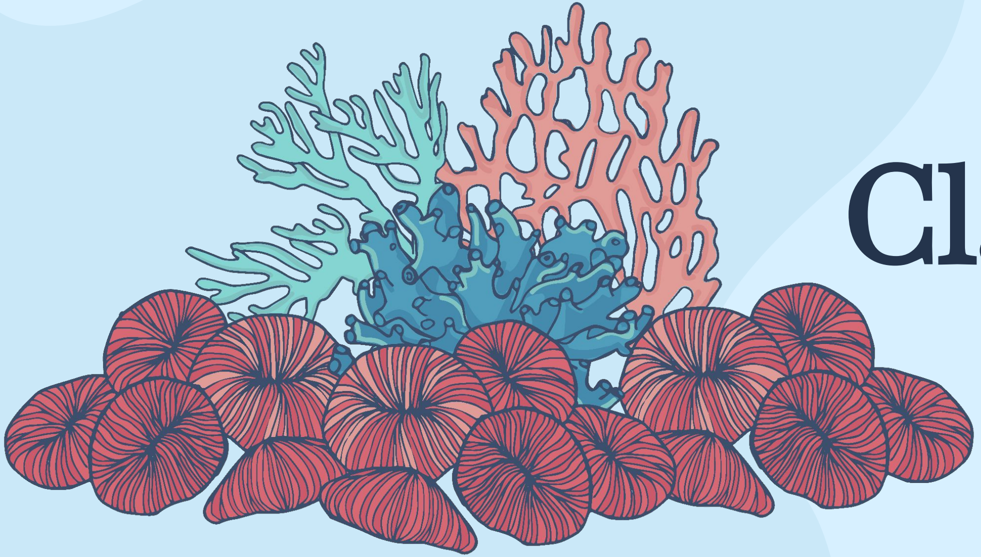
Exploratory Analysis - Fathom Net



03

Classification

Focus on the coral health status

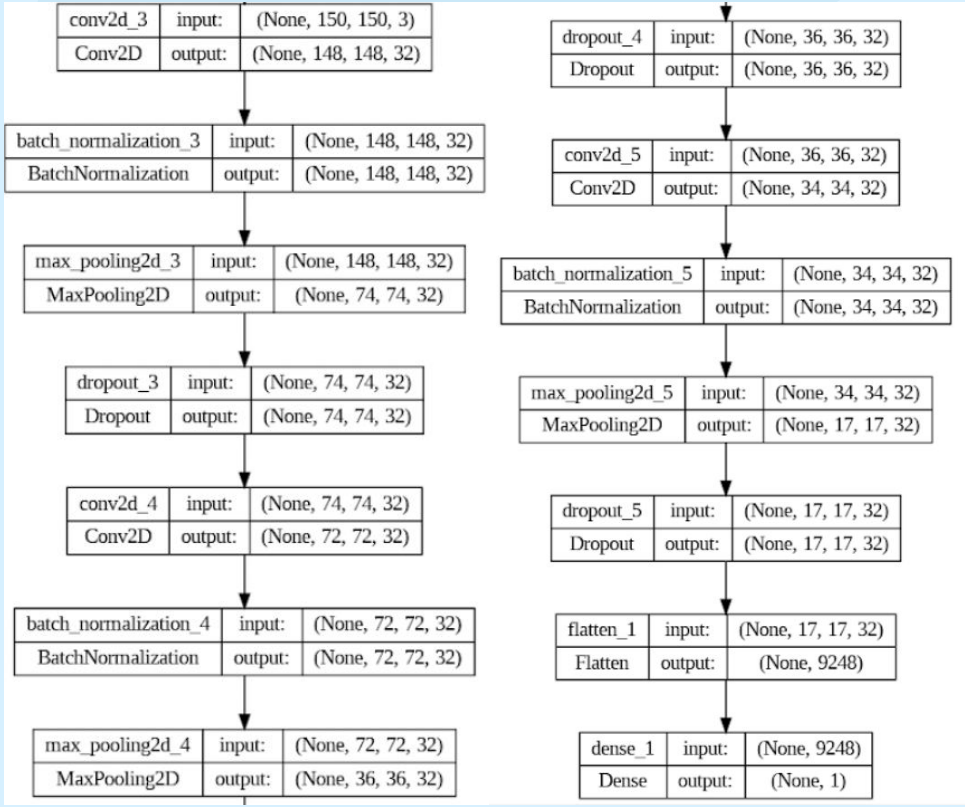


Model Selection

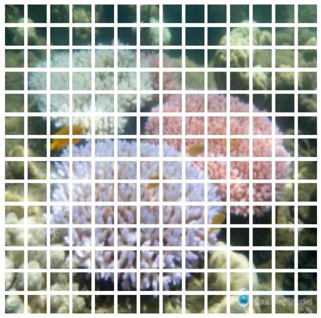
Model Name	Description
Custom	Manually constructed CNN with convolution layers, batch normalization, dropout, and max pooling.
Yolo8	Utilized the Ultralytics package for Yolo8 classification.
EfficientNet	Used pretrained EfficientNet models (b0, b1, n2) with applied dropout rates.
Vision Transformer	Employed state-of-the-art Vision Transformer model for classification.

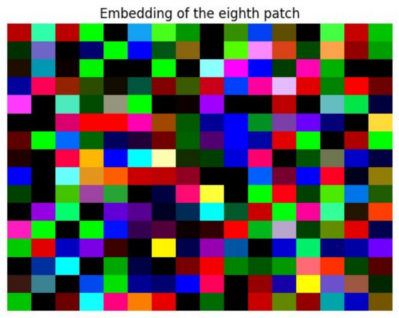
Model Architecture - Custom

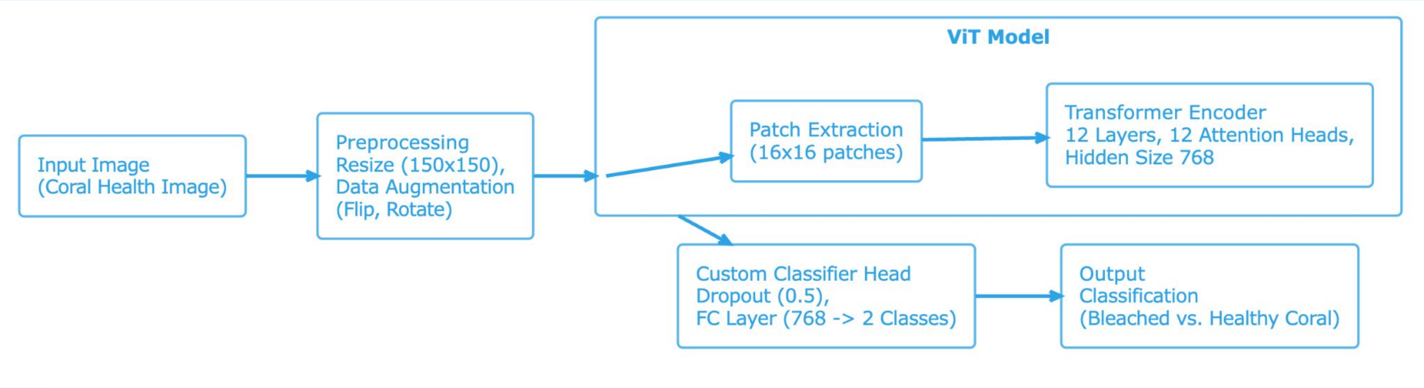
- **Input:** 3D shape (150, 150, 3)
- **Convolutional Layers**: 3 layers, 32 neurons each, ReLU activation, L1/L2 regularization
- **Batch Normalization**: After each convolutional layer
- **Max Pooling**: 2x2 pool size after each convolutional layer
- **Dropout**: 20% rate after each pooling layer
- **Flatten**: Converts 3D output to 1D
- **Output**: 1 neuron, sigmoid activation for binary classification



Model Architecture - Vision Transformer



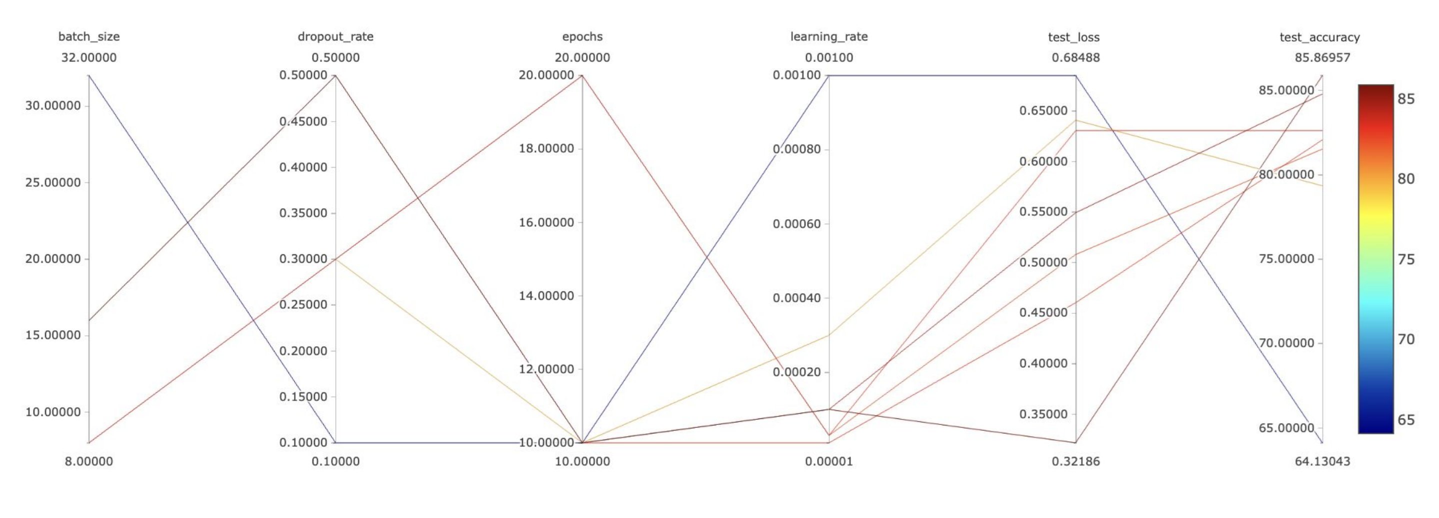




Results and Metrics

	Custom	Efficient Net	Yolo8	Vision Transformer
Test loss	0.77	0.83	0.53	0.32
Test Accuracy	61.96	47.28	78.26	85.86
Test Precision	0.79	1.0	0.89	0.84
Test Recall	0.26	0.47	0.72	0.82

Parameter Tuning - MLflow

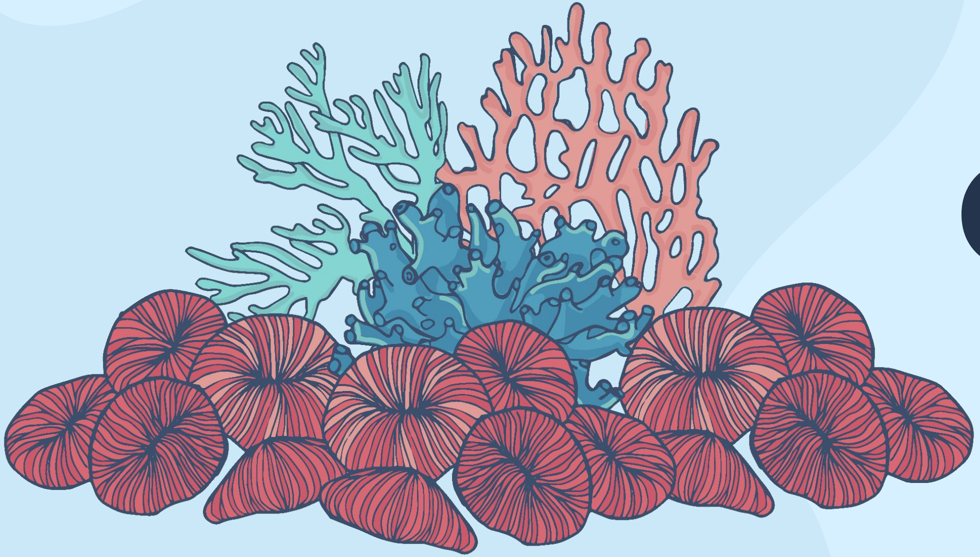


batch_size	8	8	16	32	8	16	16
dropout_rate	0.3	0.3	0.5	0.1	0.3	0.5	0.5
epochs	20	10	10	10	20	10	10
learning_rate	3e-05	0.0003	1e-05	0.001	3e-05	0.0001	0.0001
optimizer_type	AdamW	Adam	AdamW	Adam	AdamW	Adam	Adam
patience	5	5	3	3	5	3	3
weight_decay	0.0001	1e-05	0.001	0	0.0001	1e-05	1e-05

04

Generation

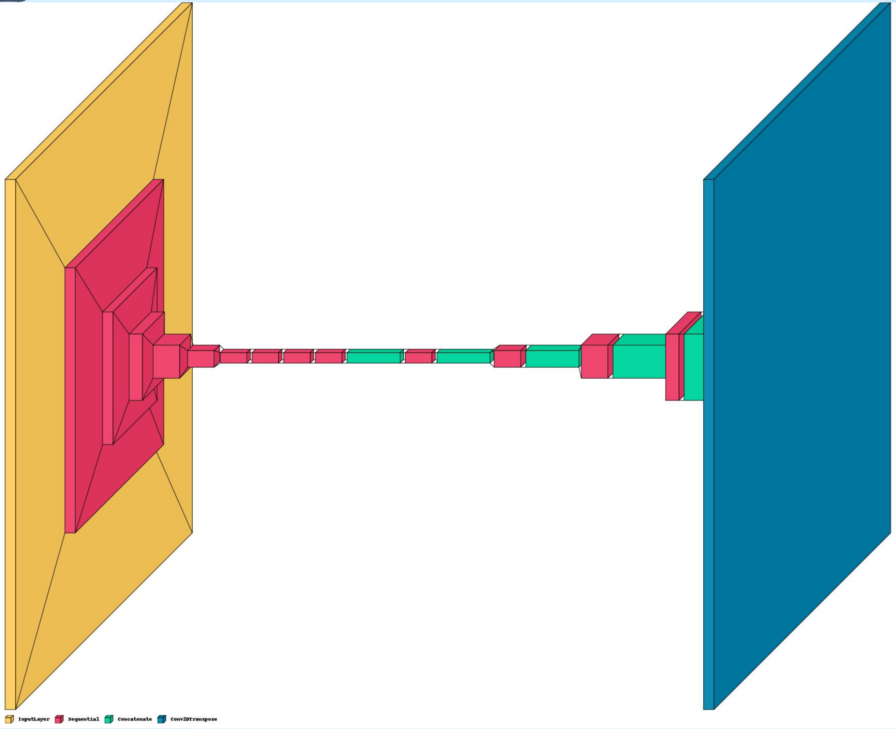
Focus on the coral health status



Model Selection

Model Name	Description
Custom CycleGAN (Coral GAN)	• By generating images of healthy scenes as they might look if polluted, researchers can better communicate the potential impact of **environmental degradation**. • Conversely, showing how degraded environments could **recover if rehabilitated** or if pollution sources are mitigated could be powerful for planning and educational purposes.
CartoonGAN	**Pretrained models** named after directors like Hayao Miyazaki, Mamoru Hosoda, Satoshi Kon, and Makoto Shinkai help transform images into **specific cartoon styles**. They provide a simple way to explore **ocean species in artistic styles** and in digital media.

CycleGAN Model Architecture



2 Generators
(healthy_generator; bleached_generator)

1. Input Layer: images of size 256x256 pixels.

2. Down-sampling Section: (Activation: LeakyReLU + Instance Normalization)

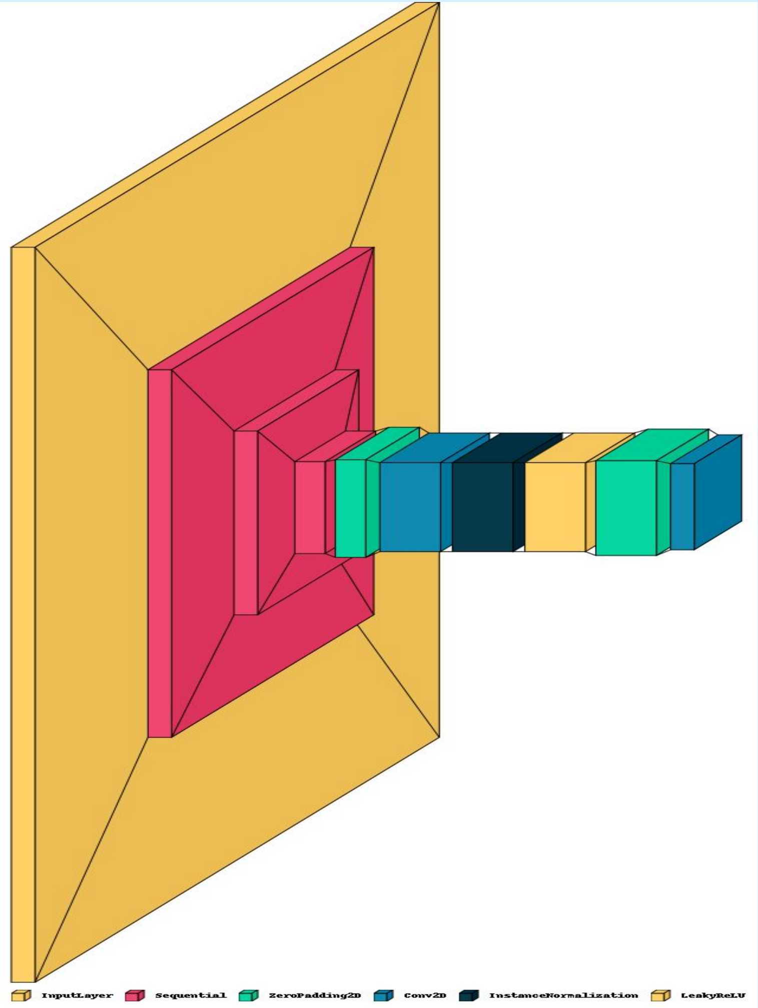
- **First Down-sampling Layer**:
  a. 64 filters, 4x4 kernel size, stride of 2, padding = 'same'.
- **Subsequent Down-sampling Layers**:
  a. Successively increases the number of filters from 128 to 512.
  b. All these layers have a 4x4 kernel size, stride of 2, padding = 'same
- **Repeated 512-Filter Layers**:
  a. Four additional layers, each with 512 filters, maintaining the 4x4 kernel size, stride of 2, padding = 'same'.

3. Up-sampling Section: (Activation: LeakyReLU + Instance Normalization)

- **First Three Up-sampling Layers**:
  a. 512 filters, 4x4 kernel size, stride of 2, padding = 'same'.
  b. Dropout applied to introduce regularization and help prevent overfitting.
- **Subsequent Up-sampling Layers**:
  a. Gradually decreases the number of filters from 512 to 64.
  b. All layers use a 4x4 kernel size, stride of 2, padding = 'same'.

4. Final Output Layer:

- **Conv2DTranspose Layer**:
  a. OUTPUT_CHANNELS = 3
  b. 4x4 kernel size, stride of 2, padding = 'same'.
  c. Kernel initialized with a random normal initializer.
  d. Activation: **Tanh** (to output pixel values in the range [-1, 1]).



2 Discriminators

(healthy_discriminatorr bleached_discriminator)

1. Input Layer: Images of size 256x256 pixels.

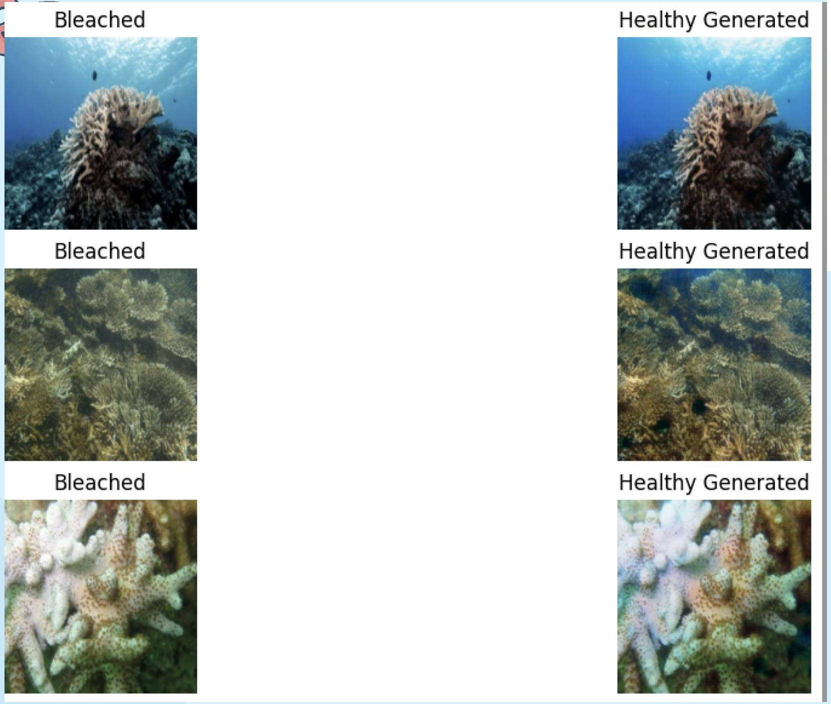
2. Downsampling Layers (Activation: Leaky ReLU + Instance Normalization)

- **First Downsampling Layer**:
  - 64 Filters, 4x4 Kernel Size, Stride of 2, no padding.
  - Reduces spatial dimensions to 128x128 pixels.
- **Second Downsampling Layer**:
  - 128 Filters, 4x4 Kernel Size, Stride of 2, no padding.
  - Reduces spatial dimensions further to 64x64 pixels.
- **Third Downsampling Layer**:
  - 256 Filters, 4x4 Kernel Size, Stride of 2, no padding.
  - Further reduces spatial dimensions to 32x32 pixels.

3. Convolution and Normalization (Activation: Leaky ReLU + Instance Normalization)

- **Zero Padding 1**: Adds padding around the edges of the feature map from the third downsampling layer.
- **Convolution**:
  - 512 Filters, 4x4 Kernel Size, Stride of 1, no padding.
- **Output Convolution:**
- **Zero Padding 2**
- **Final Convolution**:
  - 1 Filter, 4x4 Kernel Size, Stride of 1, no padding.
  - **Outputs a single-channel feature map sized 30x30.**

Examples



More in the web !

Results and Metric(**Generator Loss**, **Discriminator Loss,** Cycle Consistency Loss, Identity Loss)

Epoch	Healthy_gen_loss	Bleached_gen_loss	Healthy_disc_loss	Bleached_disc_loss
1	4.7285	4.6222	0.6395	0.6645
2	3.271	3.2022	0.6414	0.6508
3	2.8221	2.7994	0.6426	0.6358
4	2.639	2.6407	0.6384	0.6326
5	2.5603	2.5766	0.6483	0.6289
6	2.4702	2.4876	0.6426	0.6292
7	2.412	2.4596	0.6477	0.6252
8	2.3397	2.3922	0.6433	0.6233
9	2.2989	2.3386	0.6451	0.6233
10	2.2488	2.3134	0.6441	0.6137
11	2.2081	2.2835	0.6446	0.6161
12	2.1653	2.2363	0.6508	0.6207
13	2.1477	2.2109	0.6454	0.6183
14	2.1118	2.1814	0.645	0.618
15	2.0977	2.159	0.6458	0.6147
16	2.0762	2.1434	0.6446	0.6159
17	2.0692	2.1219	0.6473	0.6245
18	2.041	2.1027	0.6491	0.6213
19	2.0289	2.0748	0.6431	0.6235
20	2.0467	2.0979	0.6469	0.6237
21	1.9973	2.0327	0.646	0.6248
22	2.0206	2.0596	0.651	0.6272
23	1.9963	2.0159	0.6459	0.6296
24	1.9786	2.0005	0.6407	0.63
25	1.9859	1.9788	0.6425	0.6427

05

Discussion

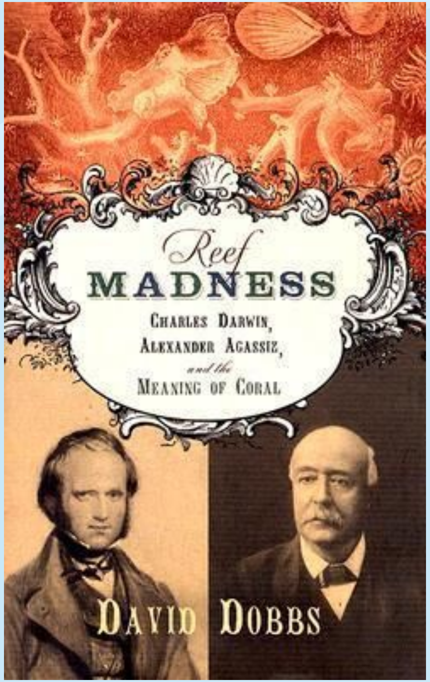
Limitation and Challenges

Limitations and Improvements

- **Large Image Size**: Large image datasets challenge computational power, especially using Fathom Net. Leveraging cloud computing resources is necessary for efficient training. Currently the models has been mainly operated using google colab to leverage the GPU power.
- **Classification Accuracy**: the healthy coral images has relatively low resolution (300 * 300 pixels) which creates challenge for improving model performance.
- **Customized Detection Model**: Achieving good Intersection over Union (IoU) results with the custom detection model has been challenging. In contrast, Yolo8 has demonstrated reliable performance without such issues.
- **GAN Model Evaluation**: Further qualitative evaluation of the GAN model is needed, using metrics such as Inception Score and Fréchet Inception Distance to assess performance.

06

Application



Experience REEF MADNESS!

Thank you!

Together let's save the planet by protecting our baby coral.